

PMT Technical Documentation

Database Maintenance

The [<http://www.sqlite.org> SQLite] database used in the PMT is a light weight and easy to use SQL database that should provide years of trouble free service. Here you will find some convenient solutions to tasks you may encounter when administrating the PMT for your users.

Updating Tables

See the page [UpdatingTables](#) for how to update tables in the PMT database.

Clearing The Database

To clear a database you empty the database tables you select of all their data. This is useful when you want to send a copy of a database to a new client. Follow these steps to clear a database of all current user data, but keep the reference codes, formulas, reporting structure, etc. for use by your new client:

1. Open Firefox
2. Select the **Tools** main menu item and select the Extension **SQLite Manager** (If the **SQLite Manager** is not present as an option, install this extension from <https://addons.mozilla.org/en-US/firefox/addon/5817>)
3. After the **SQLite Manager** extension opens, open the database you want to clear from the main menu item **Connect Database**, or from the **Recently Used** list if you opened it before.
4. Click on the **Execute SQL** tab, and paste this in:

delete from orgs;
5. Run the SQL

That should clear all user data, but will not delete the reference codes, formulas, reports, etc. that are used by all users. After you have cleared the database, you may close the **SQLite Manager**.

Vacuuming The Database

You may note that the SQLite database does not get smaller after you clear everything. This is the way databases work to be efficient during deletes: deletes do not remove data, they simply free mark the space and allow it to be over written later. To actually reduce the database size to reflect the deleted data, we must explicitly tell the database to do so by vacuuming the database. This has to be done using the SQLite3.exe version 3.6.11 program which has to be installed separately. Then you

can run this from the Windows **Run** command box: `""sqlite3" "pmt.sqlite" "vacuum;""`
Where `""` is the path to the sqlite3.exe program and the copy of the PMT database you want to vacuum. These two paths can be different of course. For example:
`""C:\Program Files\sqlite3\sqlite3" "C:\Program Files\PMT\pmt.sqlite" "vacuum;""` A black C:\ window will appear briefly, then close. If the vacuum went okay, your database size should have been reduced. A vacuum is not required. But it makes the database smaller and really deletes data if it is sensitive.

PMT Administrator Tool

The Administrator Tool is an advanced feature. It need not be delivered to all users, so is an optional file. If you remove the **admin.py** file from your PMT folder, and restart the PMT, the PMT still works, but you will notice that there is no longer a link to the Administrator Tool.

Characters Allowed in References

Most any letter and number should be okay to use as PMT formula and data entry code reference names (Regular Expressions withing the PMT should allow in theory local character sets, like Chinese, but this is currently untested). Most any other character such as `-,+/,!^`, or which is a special server character like `&,[,],{,},(,)` or a PMT control character (see below) and similar can not be used. **Allowed:** A01, B29a, CY11 **Not allowed:** (A01), +B29a, {CY11}, etc. Currently, alias are not enabled. To reference something like A01a as A01 in a report would need to add a name alias option.

Formula Types

There are two types of formulas: Those used when entering data (or **column formulas** since they only sum values within one column (ie. each month)) and **summary formulas** which are used for summary report fields and can vary in the method of summary calculation either within a row, within a column or a mix of both. To deal with this mix there are some basic rules for the summary formula field in the Administrator Tool:

1. If the field is left blank, the report will use the last value in the quarter for the summary value. This is relevant for e.g. all Balance Sheet items (end of period).
2. If the field has a formula or ref value, the summary field will use this formula as a sum formula and sum over all available periods in the quarter or year. This is relevant for all Income Statement items and some items of the Portfolio and Outreach section ("during period"). For example, include the reference field **A01** in the summary formula and if the quarter is from January to March, the report will use the formula as :

Summary A01=A01(January)+A01(February)+A01(March)

3. Normally, **summary formulas** only use data from within their own row. However, if the field formula has a reference encased in braces **{ }** then this means *"treat the formulas as a column formula, and use values from other report reference codes when calculating that summary field value"*. Thus if:

$$\mathbf{B01}=\{\mathbf{A01}/\mathbf{A02}\}$$

then the report will first calculate **A01** (as shown above as a sum of the three months), and then **A02** (using whatever formula that field has) and using these two summary values it will calculate the value for **B01**.

4. The character **?** means "make sure you calculate this value first, it is needed in the formula". So in:

$$\mathbf{A05}=\mathbf{A01}+\mathbf{?B02}$$

the **?** tells the method to first make sure the value of the **B02** field is calculated and known first before anything else is done in the formula. This is needed since reference values and their formulas are processed in order, thus all **"A"** references are processed before any **"B"** references. Thus we needed to process **"B02"** first in a formula otherwise full of **"A"** references since **B02** would not normally be calculated until after **A05**. The **?** eliminates this circular logic and forces **B02** to be calculated "out of normal order" to **A05**.

Formula Syntax

The important thing to remember is that the PMT is not a formula parser, but rather a formula "substituter". That is, the formula notations are only designed to note to the program when, where and how to replace reference codes in the formulas with actual values, and where those values come from.

Control Characters

Mathematical Operators

Plus, minus, divide, multiply, round parenthesis: + - / * () All have their regular arithmetic meaning.

COUNT

Tells the substituter to replace the key word **COUNT** with the number of actual periods with data. It is used mostly to create "average" values in each period. Ergo, if you have only two months of data in a quarter, **COUNT** will be replaced with "2".

Example:

if **COUNT=2**, then **A01=B02/COUNT** becomes **A01=B02/2**

PERIODS

Tells the substituter to replace the key word PERIODS with the number of periods in the current workbook (either 4 for quarters or 12 for months). Example:

if PERIODS=12, then A01=B02/PERIODS becomes A01=B02/12

EOY

A method of the format:

EOY(ref)

where ref is any available reference code. This method means "use the last calculated fiscal year [#FormulaTypes summary formula] value for that reference code".

LFY

A method of the format:

LFY(ref)

where ref is any available reference code. This method means *use the last manually entered fiscal year value or column formula value for that reference code*. The **LFY** differs to the **EOY** in that the **LFY** is the value last entered or calculated for the last period in the previous year for any reference code, while the **EOY** value is the final calculated summary value for the entire year that reference code.

Brackets

Brackets **[]** means "use the value from the last period". If in January **A01=1** and in February **A01=2**, and in February the system processes the formula **[A01]**, then it goes back to January (the last period) and substitutes in the value "1" for **A01**.

Braces

Braces **{}** means to calculate the summary formula as if it were a column formula instead.

At @ symbol

The @ is a code to tell the PMT formulas substituter to not do anything, but just evaluate the value "exactly as it is". This is a way to get direct access to Python arrays and other storage structures within the PMT source code. For example, there is in the PMT report class an array called **self.reportReferences**. This array can be accessed directly using the @ symbol such as:

@self.reportReferences[0]["B01"]

The array **self.reportReferences** is a two dimensional array, with the first dimension representing the period (from 0 to **PERIODS**), and the second dimension storing the reference code value for that period. So the notation `![0]!"B01"]` part of the above code is a reference to "last year's value (i.e. ![0]) for reference B01". In the program the system will evaluate the "formula" as:

```
eval('self.reportReferences![0]!"B01"]')
```

which will reference and supply the value to this array value directly. Note however that:

```
LFY(B01)=@self.reportReferences[0]!"B01"]
```

So you may also use **LFY(B01)** as an equivalent in this case. Equally, you may use this which is essentially pure Python:

```
@(self.reportReferences[self._lastQuarterColumnWithData()]!"B04"]-  
self.reportReferences[0]!"A30"])/self.reportReferences[0]!"A30"]
```

or you can reduced it to this using PMT formula syntax to:

```
([B04]-LFY(A30))/LFY(A30)
```

Both of the above options will work the same.

Ampersand: &

This is not a formula feature, it rather has special meaning for network traffic notation and can not be used for that reason.

Further References

There is some documented of formula writing in the source code (see in `reportbuilder.py`) methods: `_setSummaryColumn`, `_calculateSumPeriodUsing`, `_calculateSpecialChars`, etc.), but that is a programmers solution, not a user's solution.

Formula Correction and Deployment

The concept is simple but not fully implemented. That is, send out an SQL update which the user applies. For example to update the formula for reference SAV2 one might use:

```
update field_references  
set attributes='sum_period_using="(A16b+A17b)/A15"  
where ref='SAV2'
```

Delivery is another issue. Possible options for delivery:

1. An "Install" packages can be built to do this.

2. Automate it (the system auto links to a valid Internet site and system auto processes the update). Clearly there should also be an automatic backup before any such updates are applied.

For any solution not hand delivered there needs to be a secure server to link for downloads or "Mr. Man in the Middle" can redirect a

delete from

and erase all the data.

PMT Database Structure

To do.

PMT Technical Documentation

Hardware Requirements

A 1 GHz processor with at least 1 MB of RAM recommended and 120MB of hard disk space (70 MB for Python, 40 MB for Firefox, and 10 MB for a PDF viewer). The PMT can be installed on smaller Netbook computers, so the hardware requirements are quite modest.

Software Operating System

Windows, Linux, Mac, BSD or other operating systems that has or can install Python and a W3C compliant web browser.

Installation

Windows

Installation may be done using the official PMT installation program for Windows, or by doing a manual installation in the following instructions for **Other OS** below

Other OS

1. Download and install [<http://www.python.org/download/releases/2.5.4/> Python-2.5.4]. Python 3.x will not work correctly with the PMT.

Python 2.5.4 comes with an outdated version of SQLite. You will need to update [<http://www.sqlite.org/> SQLite] to version: 3.6.13 or later and install the sqlite libraries into the Python Library path ("sqlite.dll" and "sqlite.def" into **C:\Python25\DLLs** on Windows). Version 3.6.13 is stable and updating to later

versions are optional. If you do use a later version, you may need to update the PMT database format to that later version.

2. Download and install the latest version of [<http://www.cherrypy.org/wiki/CherryPyInstall> CherryPy] that will work with Python 2.5.4
3. Download and install the python [<http://pypi.python.org/pypi/minixsv/0.7> XML schema validator]
4. A W3C compliant browser such as [<http://www.mozilla.com/firefox> Firefox] is required. **Note: Internet Explorer will not work.**
5. Download a PDF viewer if you do not have one such as the [<http://www.foxitsoftware.com> Foxit Reader]
6. Download the latest PMT Source and unzip.
7. Create a script file for your OS that will start the PMT server using the correct version of Python you installed. For example, if you installed Python into the folder **c:\python25** and unzipped the PMT source files into the folder **C:\Program Files\pmt** for Windows, enter this into Notepad, and save as **pmtstart.cmd: start /d "C:\Program Files\pmt" /min c:\python25\python pmt.py** Double click on the saved **pmtstart.cmd** file to start the PMT server. You can also right click on the file and send a shortcut to your desktop for easy access. To stop the PMT Python server, click on the minimized Python program in your Windows Task Bar which appeared when you ran the above batch file, and press the key combination: **Ctrl-C**. A complete script file to start the PMT on Windows is provided as "pmtstart.cmd".
8. After you have started the PMT server in step (7) above, start the Firefox Web Browser and then open the page: **http://localhost:6711/** then login or create an account if this is the first time you started the PMT. Bookmark this page for simple access later.

Database Structure

Database Structure and Information is available at the [[PmtDatabaseStructure database structure](#)] page.

Database Maintenance

Information about database maintenance is available at the [[DatabaseMaintenance Database Maintenance](#)] page

Python Source Code

Information about the source code and Python programming for the PMT is available on the [[PmtSourceCodeDocs source code documents](#)] page.

Administrator Tool

See the [HelpAdministratorTool Administrator Tool] help section for more information about how to use the PMT Administrator Tool.

Creating an Installation CD-ROM

See the [CreatingInstallCd Creating an Installation CD-ROM] page for detailed instructions on how to do this.

PMT Technical Documentation

Hardware Requirements

A 1 GHz processor with at least 1 MB of RAM recommended and 120MB of hard disk space (70 MB for Python, 40 MB for Firefox, and 10 MB for a PDF viewer). The PMT can be installed on smaller Netbook computers, so the hardware requirements are quite modest.

Software Operating System

Windows, Linux, Mac, BSD or other operating systems that has or can install Python and a W3C compliant web browser.

Installation

Windows

Installation may be done using the official PMT installation program for Windows, or by doing a manual installation in the following instructions for **Other OS** below

Other OS

1. Download and install [<http://www.python.org/download/releases/2.5.4/> Python-2.5.4]. Python 3.x will not work correctly with the PMT.

Python 2.5.4 comes with an outdated version of SQLite. You will need to update [<http://www.sqlite.org/> SQLite] to version: 3.6.13 or later and install the sqlite libraries into the Python Library path ("sqlite.dll" and "sqlite.def" into **C:\Python25\DLLs** on Windows). Version 3.6.13 is stable and updating to later versions are optional. If you do use a later version, you may need to update the PMT database format to that later version.

2. Download and install the latest version of [<http://www.cherrypy.org/wiki/CherryPyInstall> CherryPy] that will work with Python 2.5.4
3. Download and install the python [<http://pypi.python.org/pypi/minixsv/0.7> XML schema validator]

4. A W3C compliant browser such as [<http://www.mozilla.com/firefox> Firefox] is required. **Note: Internet Explorer will not work.**
5. Download a PDF viewer if you do not have one such as the [<http://www.foxitsoftware.com> Foxit Reader]
6. Acquire the latest PMT Source tree and unzip.
7. Create a script file for your OS that will start the PMT server using the correct version of Python you installed. For example, if you installed Python into the folder **c:\python25** and unzipped the PMT source files into the folder **C:\Program Files\pmt** for Windows, enter this into Notepad, and save as **pmt_start.cmd: start /d "C:\Program Files\pmt" /min c:\python25\python pmt.py** Double click on the saved **pmt_start.cmd** file to start the PMT server. You can also right click on the file and send a shortcut to your desktop for easy access. To stop the PMT Python server, click on the minimized Python program in your Windows Task Bar which appeared when you ran the above batch file, and press the key combination: **Ctrl-C**.
8. After you have started the PMT server in step (7) above, start the Firefox Web Browser and then open the page: **<http://localhost:6711/>** then login or create an account if this is the first time you started the PMT. Bookmark this page for simple access later.

Database Structure

Database Structure and Information is available at the [[PmtDatabaseStructure database structure](#)] page.

Database Maintenance

Information about database maintenance is available at the [[DatabaseMaintenance Database Maintenance](#)] page

Python Source Code

Information about the source code and Python programming for the PMT is available on the [[PmtSourceCodeDocs source code documents](#)] page.

Administrator Tool

See the Administrator Tool help section for more information about how to use the PMT Administrator Tool.